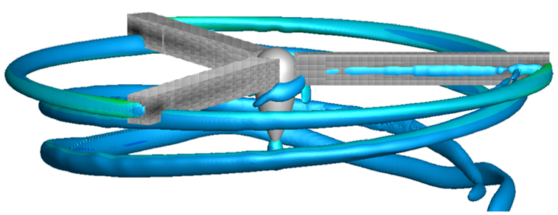
Rapid Computational Aerodynamic Analysis for Multi-Rotor Aircraft

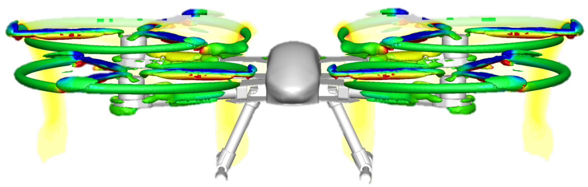
Jonathan Chiew

Advanced Modeling & Simulation (AMS) Seminar Series

NASA Ames Research Center



October 27, 2020



Computational Aerosciences Branch
Ames Research Center

Stanford ENGINEERING
Aeronautics & Astronautics

Motivation

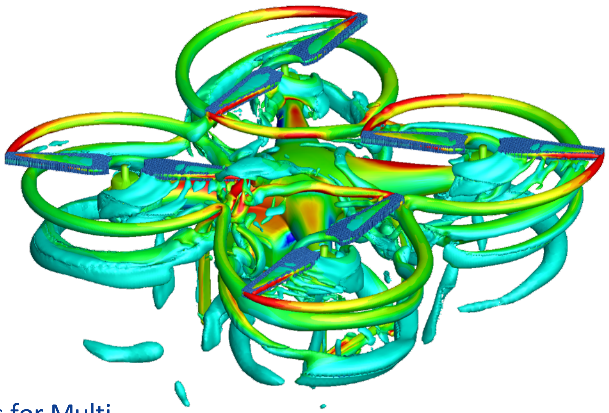

Lilly Gipson
National Aeronautics and Space Administration
NASA
CENTURY MEDICAL CENTER
www.nasa.gov

Motivation

Novel vehicle designs leveraging distributed electric propulsion and advancements in electric motors, batteries, and controllers

Multiple propeller wakes and complex aircraft geometry that generate intricate aerodynamic interactions

Shorter design cycles and budgets than are typical for traditional rotorcraft design



Outline

Introduction and research objective

Rotor modeling

Robust & performant algorithms

Concluding remarks

Acknowledgments

Research Objective

Develop a computational aerodynamic analysis tool directed towards multi-rotor aircraft and suitable for preliminary design

Principal requirements:

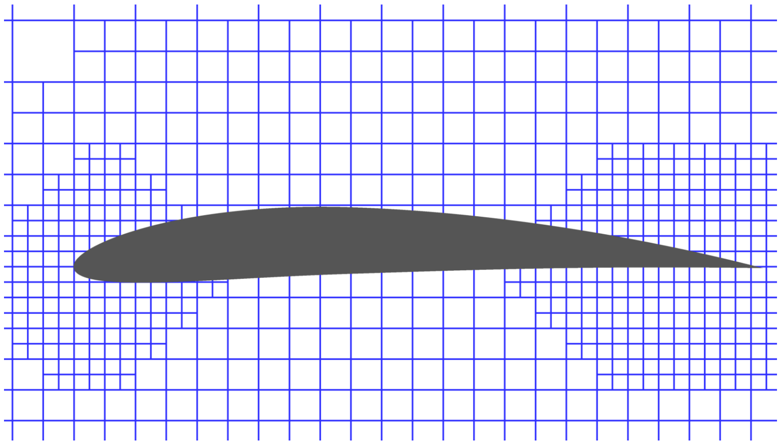
- Handle complex geometry
- Accurate rotor performance
- Turnaround time including mesh generation and flow solution <1 day on modest compute resources
  - 32-64 cores (<1000 CPU core-hours)
  - 128-256GB memory

Geometric Models

CAD model of aircraft outer mold line (OML) is typically available during preliminary design

Cartesian mesh approaches well suited for this task (Cart3D)

- OML represented as set of triangles with general treatment of both lifting and non-lifting components
- Hexahedral volume mesh with embedded boundaries robustly and automatically generated
- Excellent scalability on multi-core CPUs with domain decomposition
- Challenges in capturing boundary layer viscous effects without body-conforming mesh system



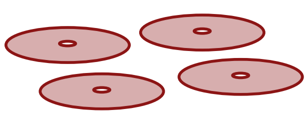
Rotor Modeling

Spectrum of Rotor Models

High fidelity three-dimensional Navier-Stokes solutions are too expensive, requiring ***O***(10,000) core-hours per simulation

Low fidelity models depend on calibrated correction factors and may not accurately predict aerodynamic performance

Blade element theory (BET) methods often use simplified fluid dynamics and surface modeling but can struggle with complex aerodynamic interactions

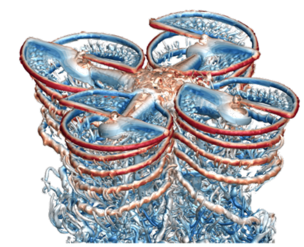


Uniform actuator disk

Blade Element Theory (BET) methods

lower fidelity
lower cost

higher fidelity
higher cost

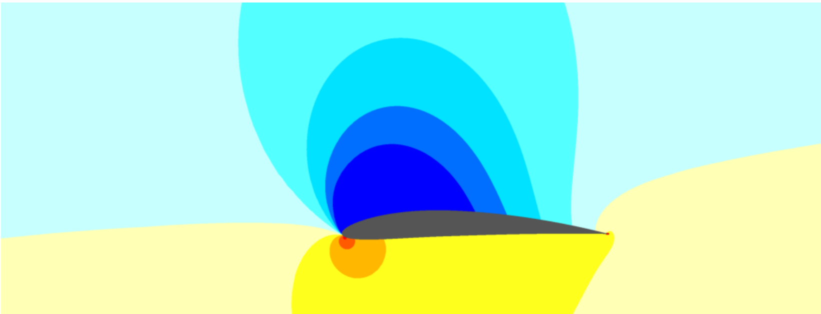


Blade resolved CFD

Blade Element Theory Models

Each propeller blade is partitioned into distinct elements in the radial direction

Spanwise (radial) effects on individual blade elements are assumed to be negligible resulting in locally two-dimensional aerodynamics



Using the corresponding freestream conditions, each blade element's aerodynamic forces can be computed on the fly or interpolated from pre-tabulated data

Lingering Open Question

Existing unsteady body force rotor models coupled to Cartesian solvers have to date demonstrated an inability to accurately predict rotor performance

- RotCFD simulations found that the unsteady rotor model underpredicted figure of merit to the point where it was simply replaced by the steady model (Koning, 2016)
- ROAM overpredicted figure of merit for a tiltrotor in hover, reaching unphysical values (Wissink et al., 2020)
- Initial unsteady Cart3D rotor model simulations were unable to accurately predict performance (Chiew & Aftosmis, 2018)

How can we attain accurate rotor performance predictions?

Determining Freestream Velocity

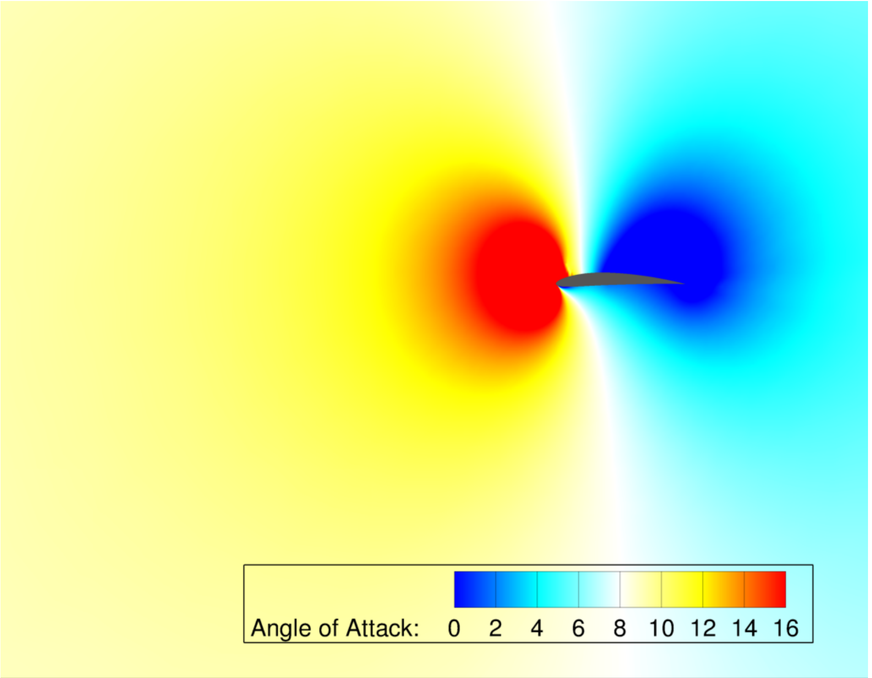
Accurate airfoil forces and moments can be readily computed once the geometry and freestream flow conditions are specified

Freestream velocity vector specification:

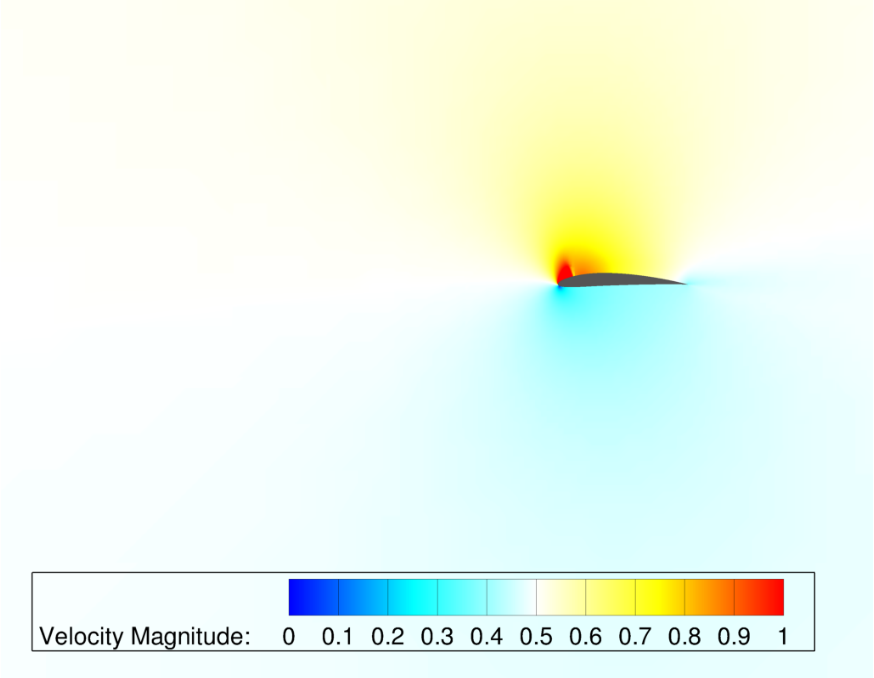
- Vector magnitude (Mach number)
- Angle of rotation to the airfoil chordline (angle of attack)

Determining Freestream Conditions

Angle of Attack

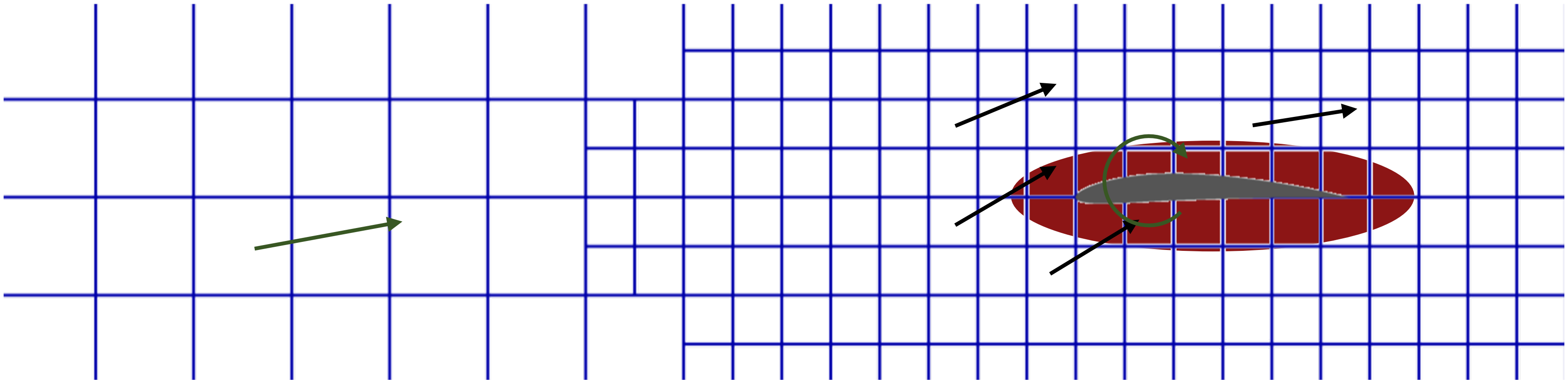


Velocity Magnitude



Correct freestream value shown in **white**

Determining Freestream Velocity



Local velocity at each cell (Rajagopalan 1991)

Reference point (optional correction) (Shen 2007, 2009)

Integral velocity sampling (Spalart 2015, 2017)

Integral Velocity Sampling

Define the airfoil body force vector in steady, incompressible flow with an arbitrary force projection function (FPF), $g\left(\mathbf{x}\right)$:

$$\mathbf{f} = -\rho \frac{c}{2} \hat{U} \left(C_l \mathbf{e}_z \times \mathbf{U} + C_d \mathbf{U} \right) g\left(\mathbf{x}\right)$$

Take the curl of the inviscid momentum equations with this body force and equate the circulation of the flow with that from the Kutta-Joukowski theorem which yields

$$\mathbf{U}_\infty = \iint g\left(\mathbf{x}\right) \mathbf{U} d\mathbf{x}$$

under the assumptions of open streamlines, zero airfoil drag, and irrotational flow upstream

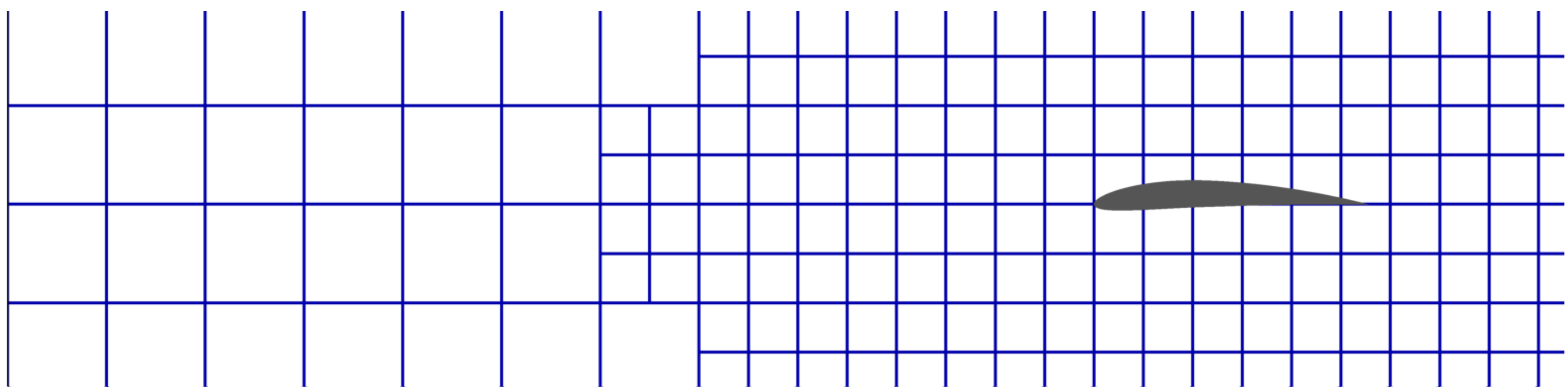
(Spalart 2015, 2017)

Integral Velocity Sampling (IVS)

The force projection function (FPF) is any vector function with compact support that integrates to unity

The freestream velocity is the FPF-weighted integral of the local velocity

This is the source area's mean velocity for uniform FPF



Integral Velocity Sampling Verification

IVS has been used in the modeling of both helicopter rotors (Forsythe, 2015) and wind turbines in CFD simulations (Churchfield, 2017)

Merabet and Laurendeau (2019) found IVS to be accurate in predicting freestream velocity but only tested it in a decoupled manner (fixed lift and drag coefficients)

Several open questions were investigated in this work:

- Accuracy of fully coupled freestream velocity predictions
- Sensitivity to the extent and choice of projection function
- Extension to three dimensions with rotating wings

IVS: 2-D Simulations

Finest mesh: 123k cells, 0.6% chord

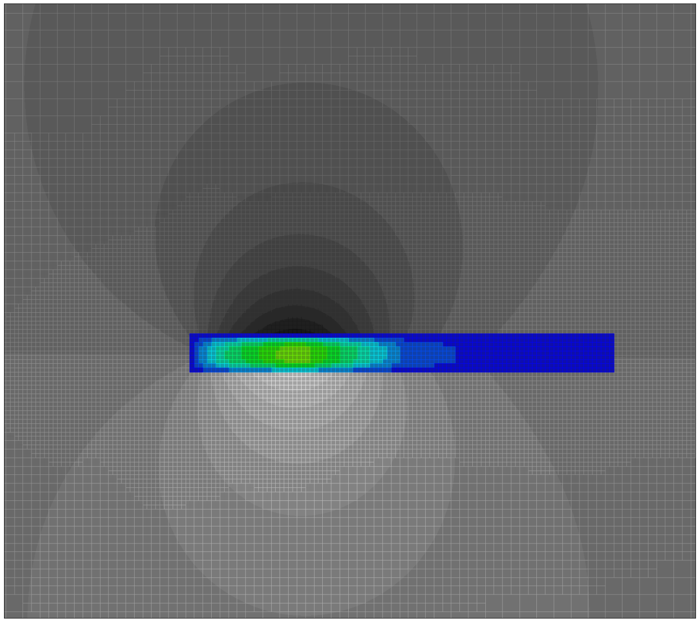
Refinement based on adjoint driven airfoil solution

Far-field boundaries: 250 chords

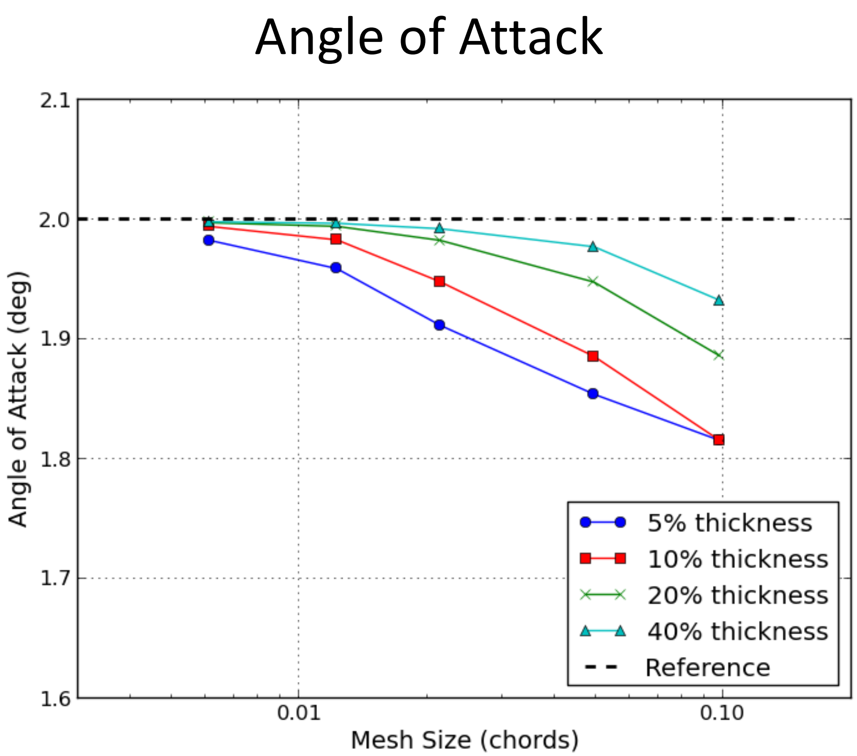
Mach number: 0.2

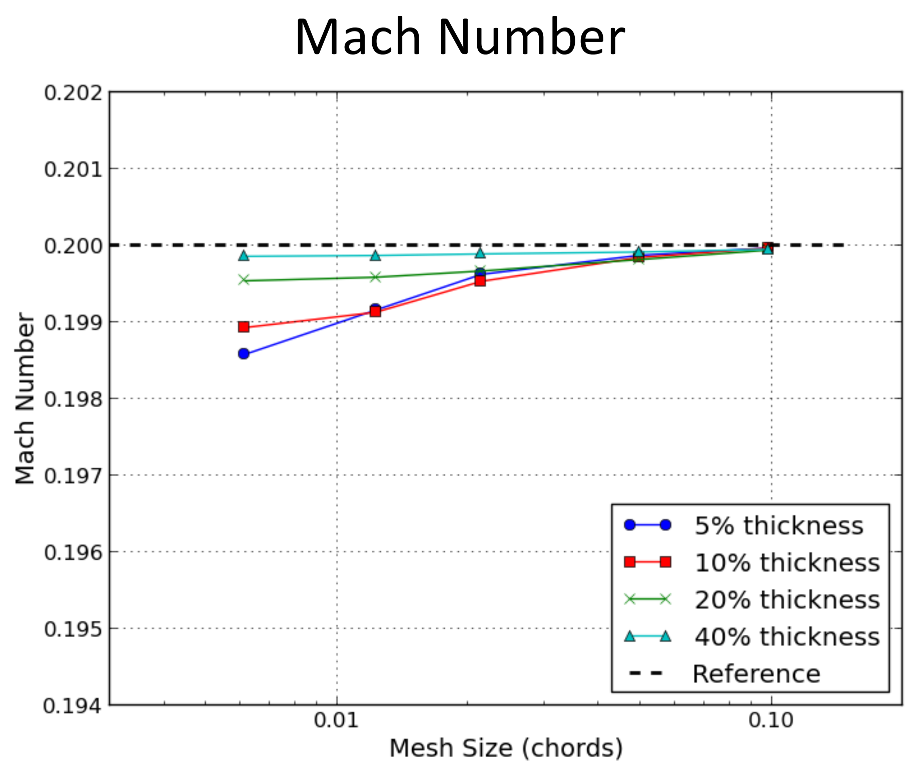
Angle of attack: 2 deg

Airfoil: NACA 0015



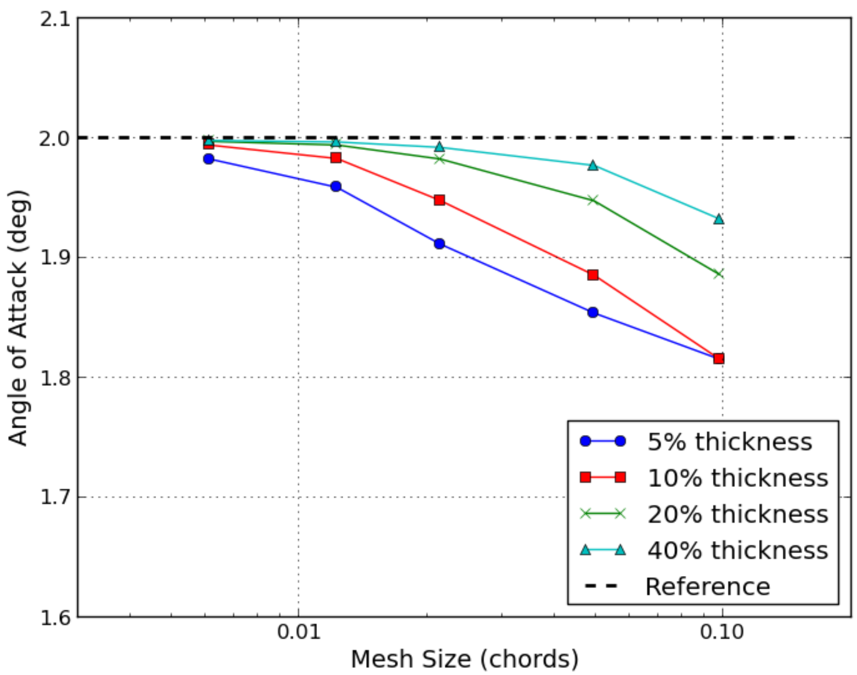
IVS 2–D Accuracy: Zero Drag



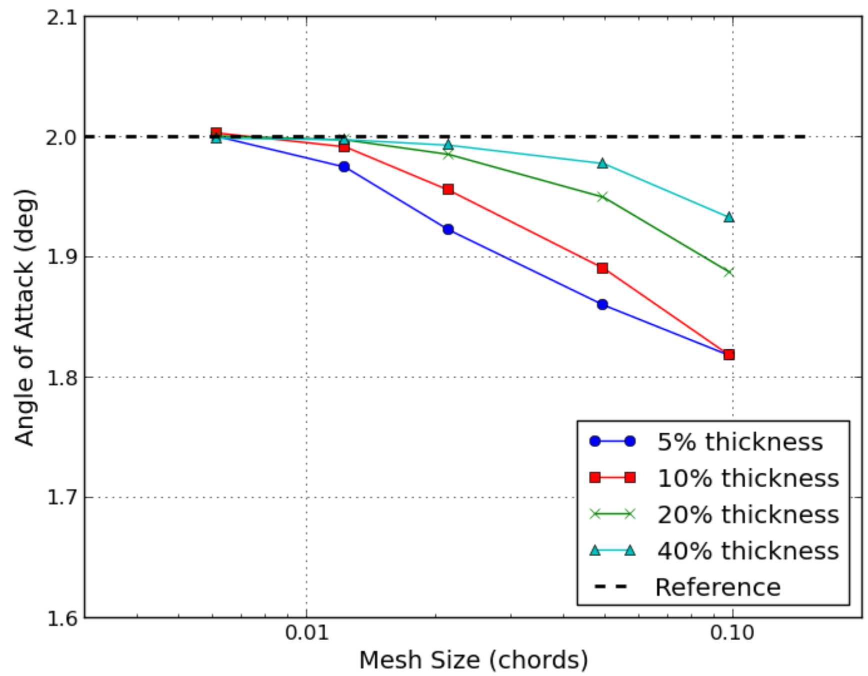


IVS 2-D: Effect of Airfoil Drag

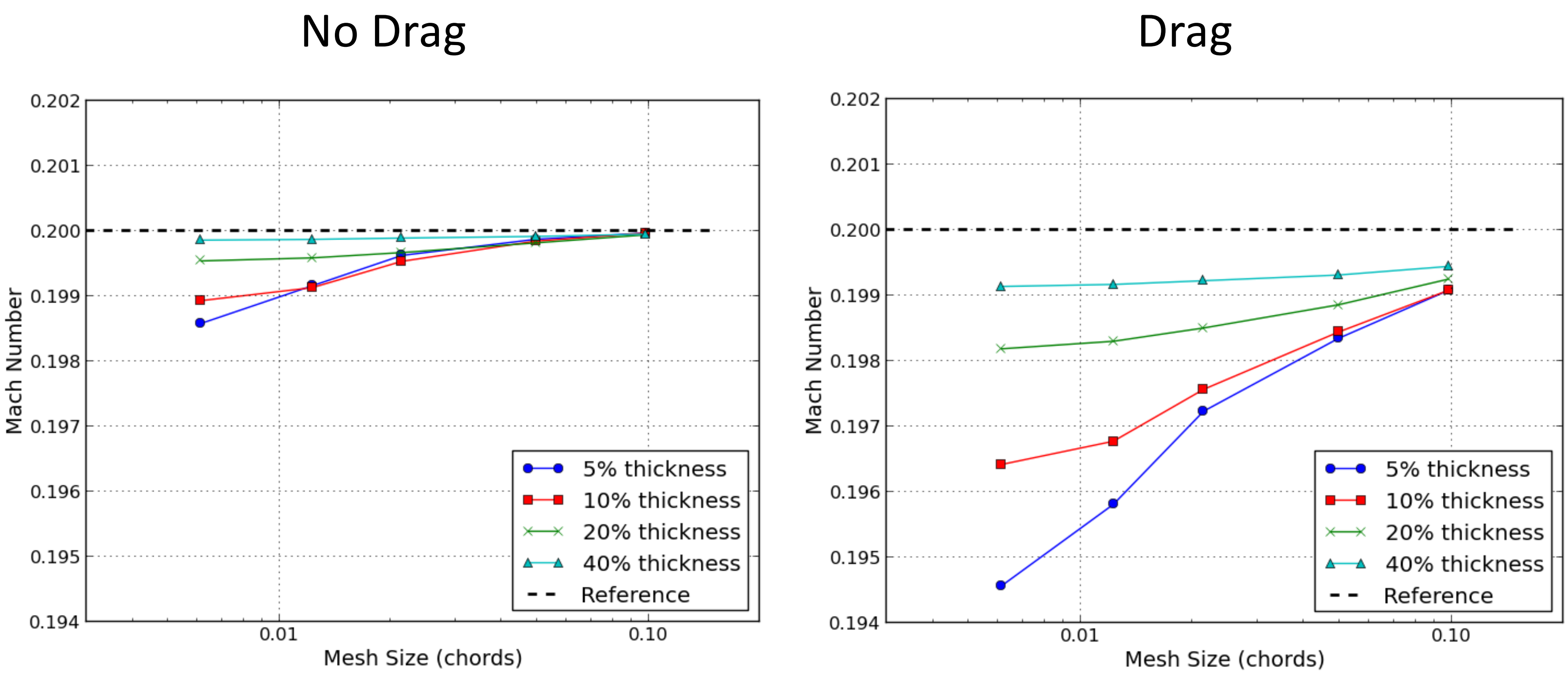
No Drag



Drag

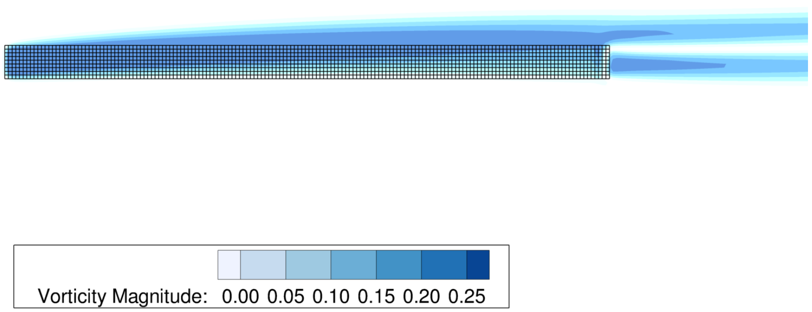


IVS 2-D: Effect of Airfoil Drag

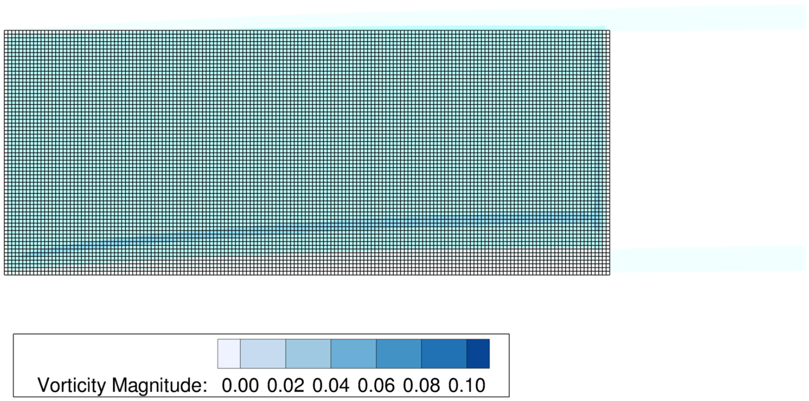


IVS 2-D: Source Region Thickness Effects

5% thickness, Δx = 0.6%c



40% thickness, Δx = 0.6%c



IVS 2-D: Effect of Projection Function

IVS derivation is independent of the projection function, but nearly all implementations use spherical Gaussian functions

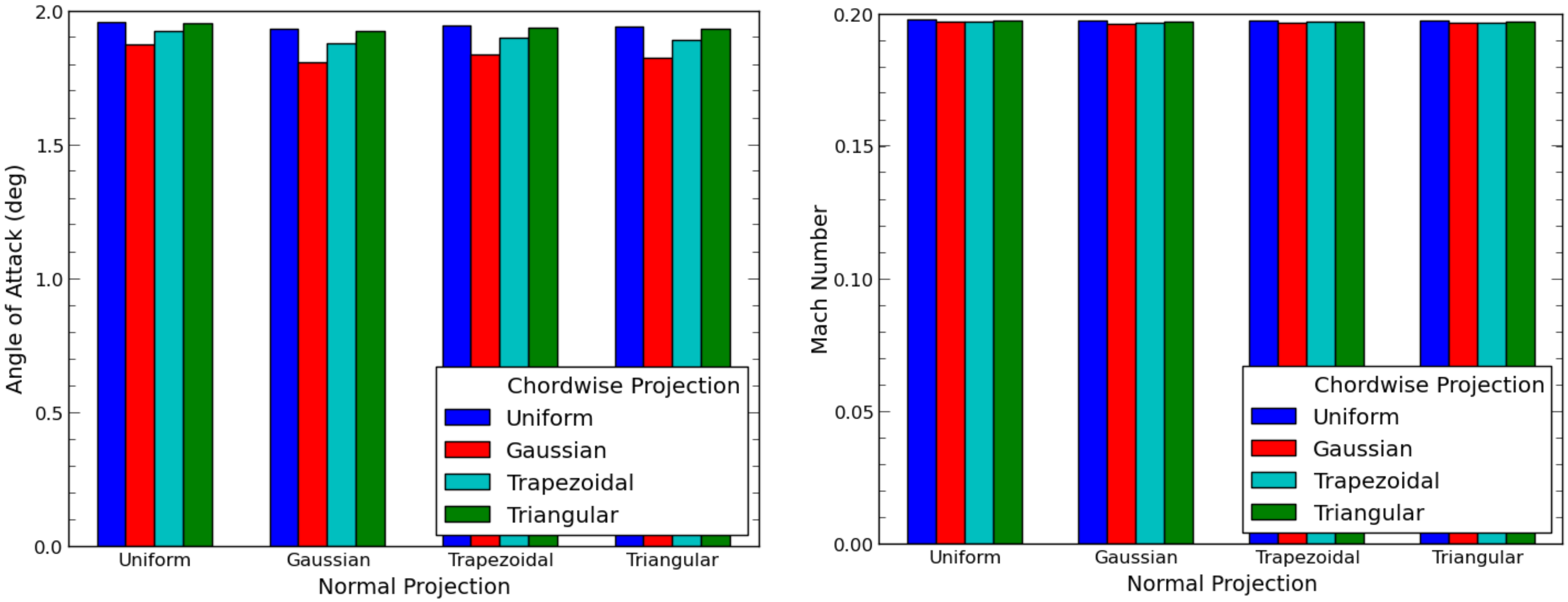
Explore the sensitivity of IVS to a variety of projection functions of the form

$$g\left(\mathbf{x}\right) : \mathbb{R}^{\mathrm{n}} \rightarrow \mathbb{R} = \prod_{i}^{\mathrm{n}} g_i\left(x_i\right)$$

multiplying functions of orthogonal coordinates

Consider uniform, trapezoidal, and triangular FPF

IVS 2-D: Effect of Projection Function



IVS 2-D: Observations

Integral velocity sampling naturally and accurately predicts freestream velocity vector

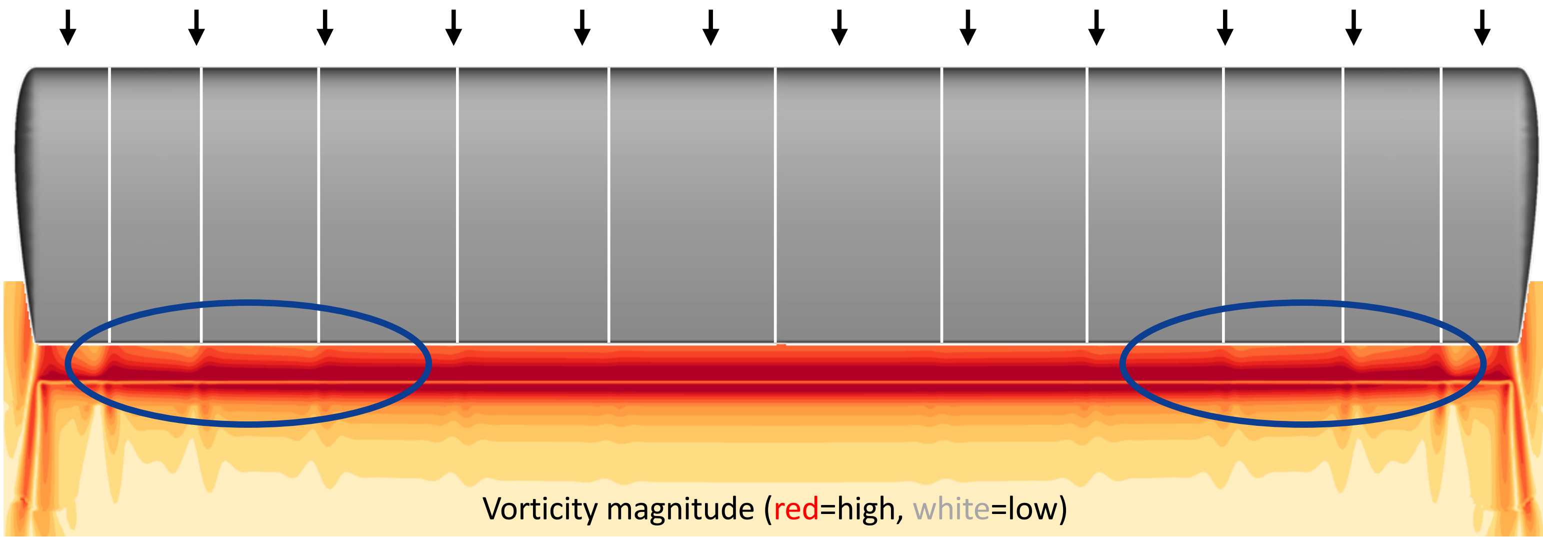
- <10% error even on coarse meshes
- Mach number prediction is an order of magnitude more accurate than velocity direction
- Generally insensitive to choice of force projection function

No additional modeling of self-induced velocity required

- Avoids secondary iterations
- Eliminates spatial or temporal offsets of a reference point

IVS: 3-D Extension

In Blade Element Theory (BET), the wing is modeled as a set of independent wing sections in two-dimensional flow

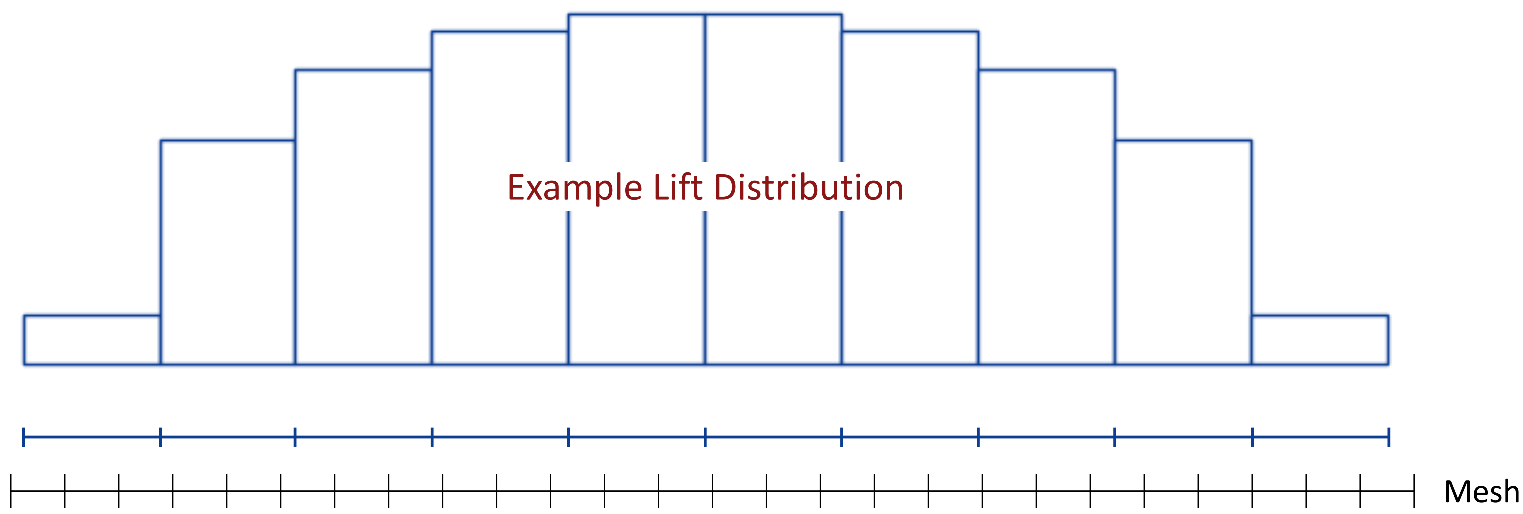


However, spurious vorticity is created at the section boundaries

IVS: 3-D Extension – Quadratic Interpolation

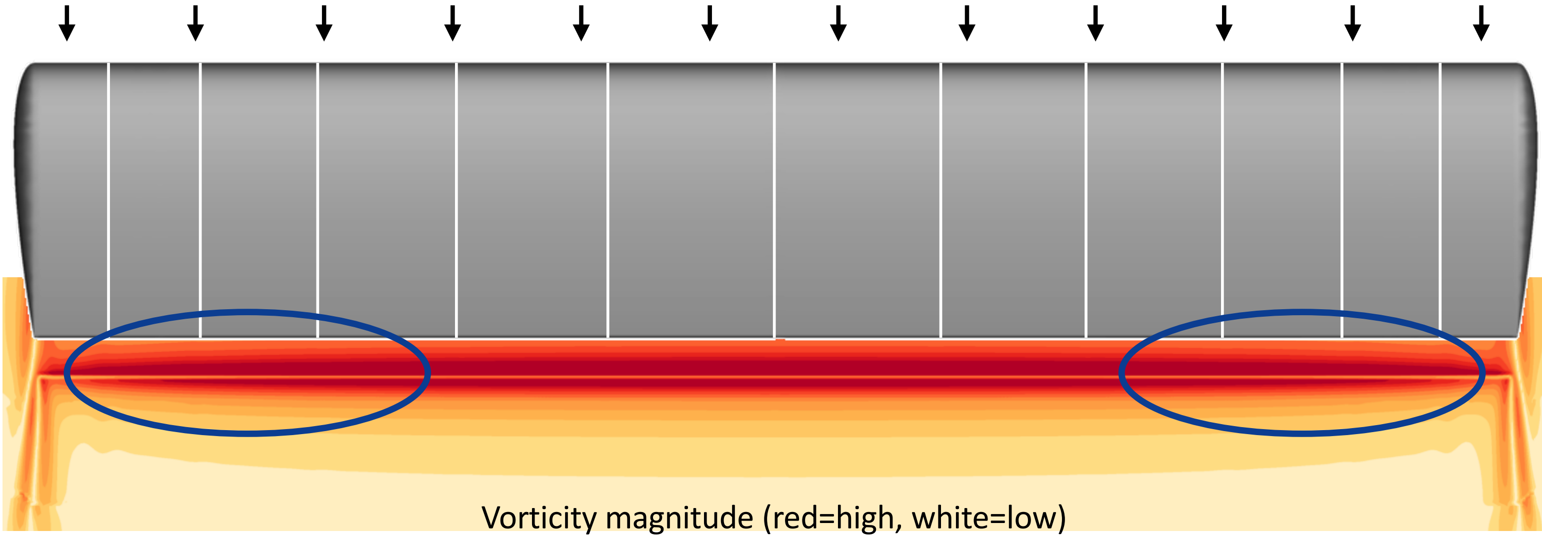
Wing lift distribution is piecewise constant over each element and sudden changes in lift cause spurious vorticity

Use quadratic interpolation and linear extrapolation to create (C^0) continuous variation of angle of attack



IVS: 3-D Extension – Quadratic Interpolation

Quadratic interpolation smooths the vorticity distribution

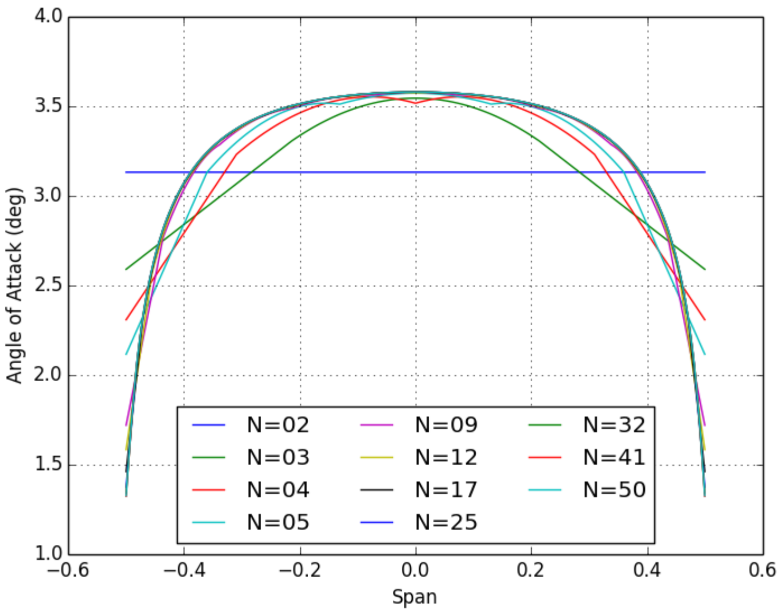


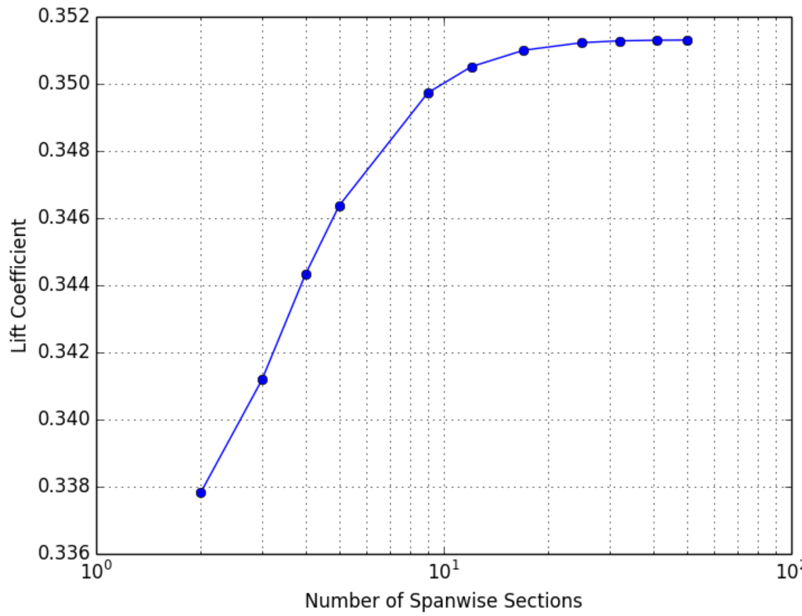
How accurate is the spanwise loading?

IVS: Spanwise Interpolation Convergence

Rectangular, untwisted wing planform

NACA 0015 airfoil, AR = 10



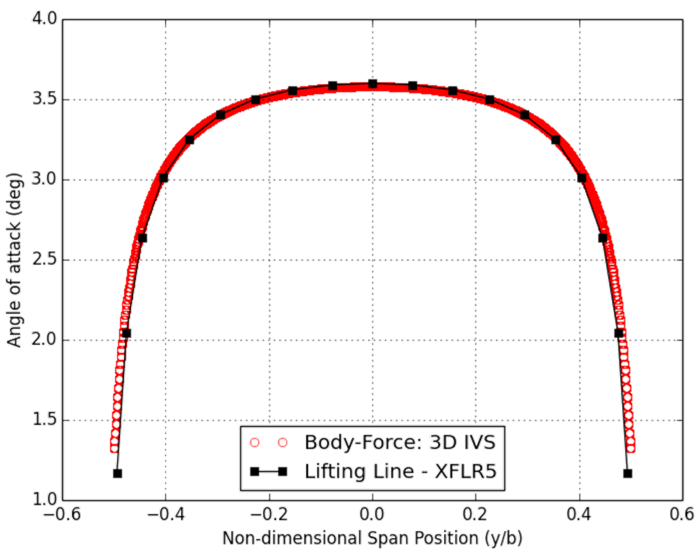


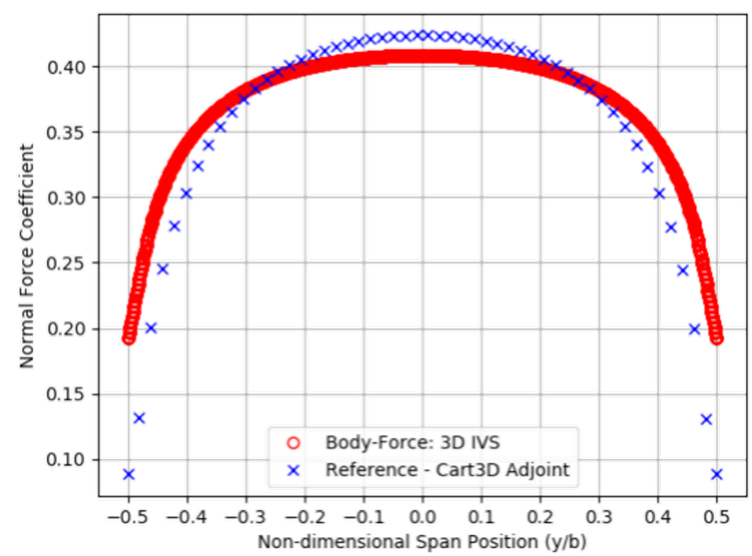
N = # of spanwise sections

IVS: Spanwise Loading

Rectangular, untwisted wing planform

NACA 0015 airfoil, AR = 10

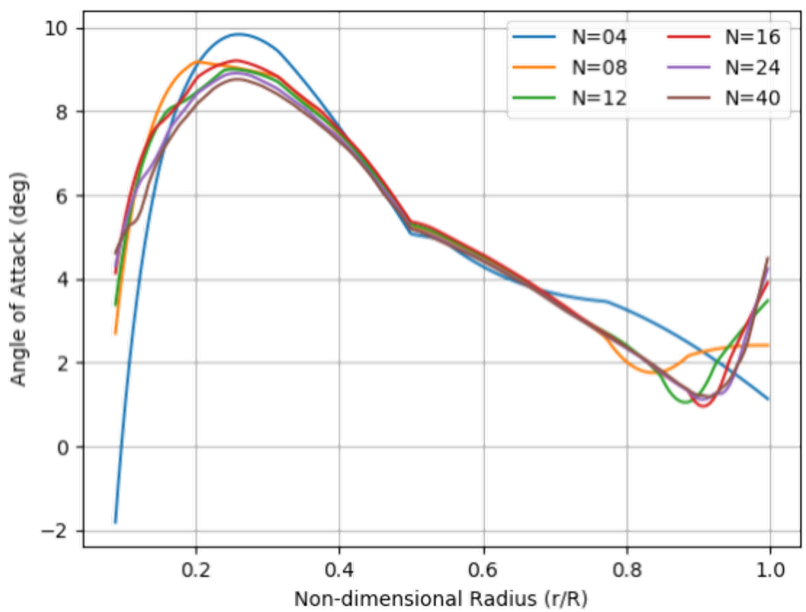


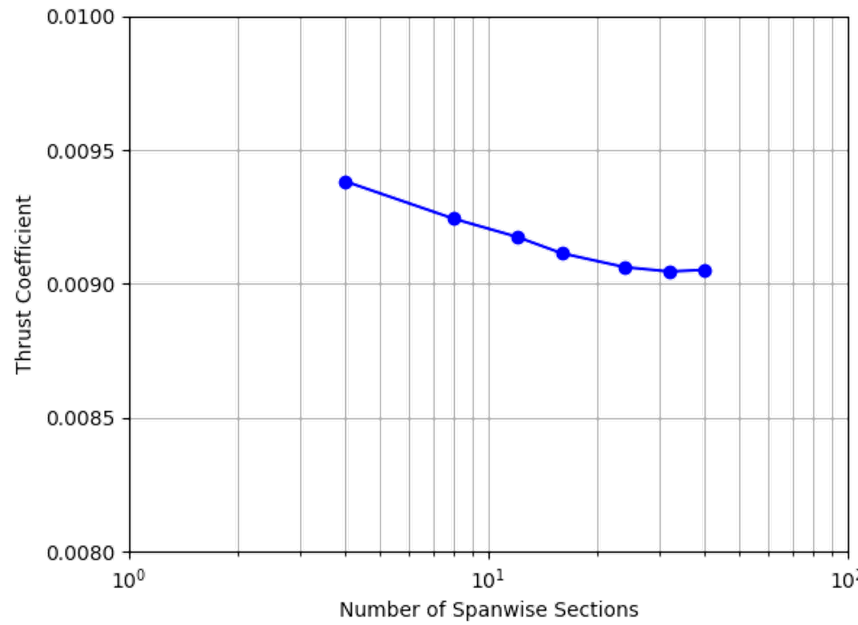


IVS: Rotating-wing Extension

Propellers have azimuthal velocity variation due to rotation

Separate quadratic interpolants for induced and azimuthal velocities in the radial direction



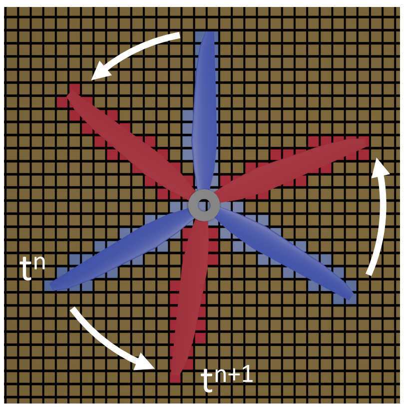


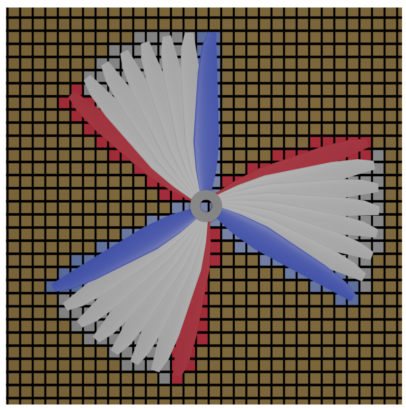
N = # of spanwise sections

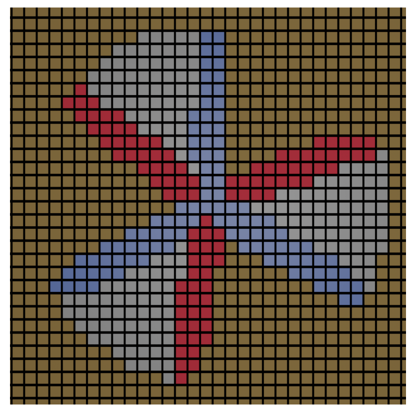
Large Timestep Extension

Typical models impose timestep restrictions so that the blade does not traverse multiple mesh elements in one step

Conversely, A-stable implicit time integration methods are often utilized to enable arbitrarily large timesteps while maintaining numerical stability



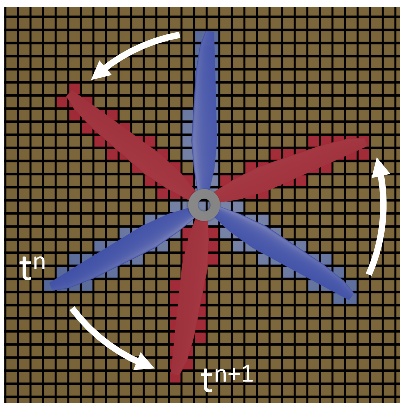


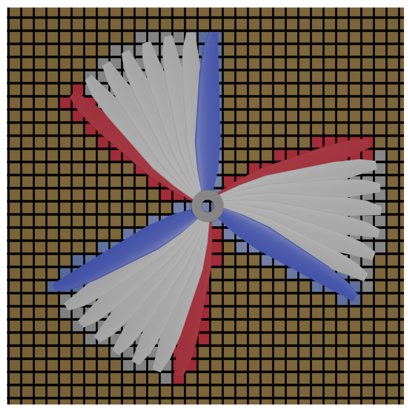


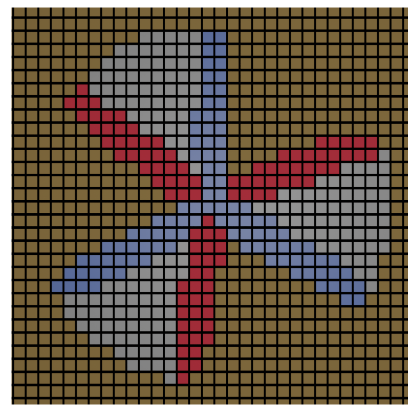
Large Timestep Extension

Total impulse imparted to each mesh element should be independent of Δt and can be controlled with azimuthal FPF

- Uniform distribution has $g_\psi \propto 1/\Delta t$ which conserves the total impulse applied
- When $\Omega \Delta t = 2\pi/N_b$ the model reverts to the steady-state formulation found in literature

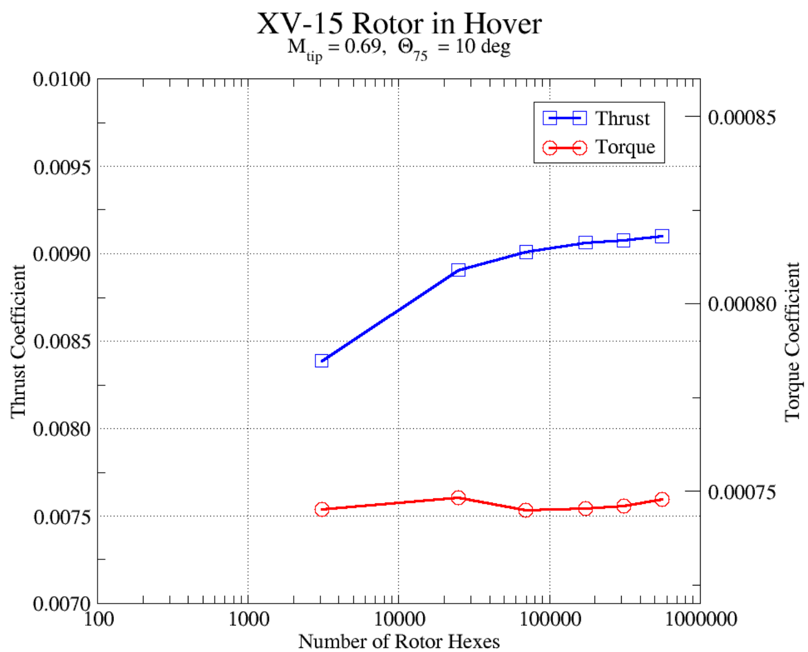




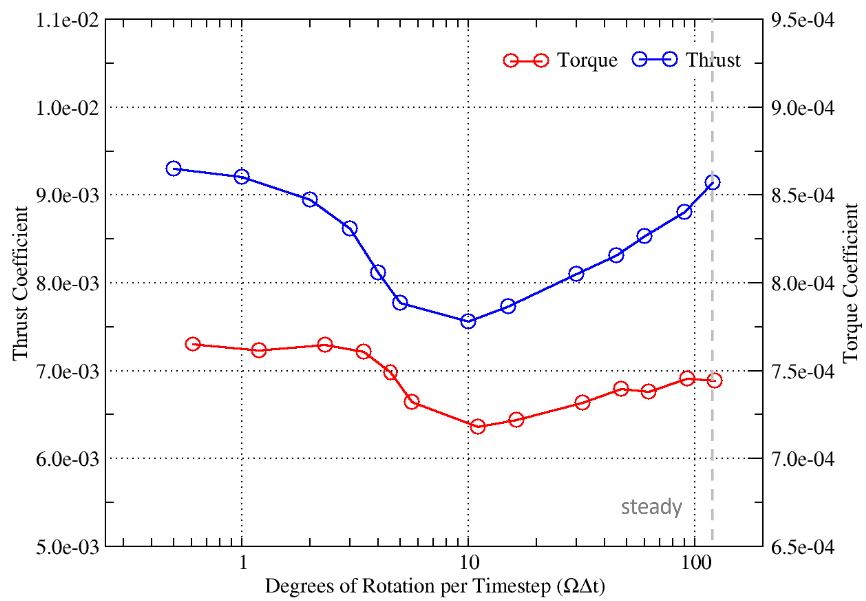


Rotor Model Verification

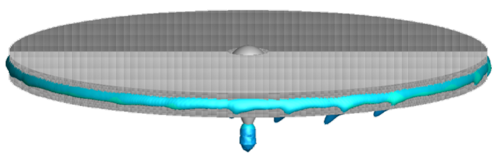
Mesh Convergence



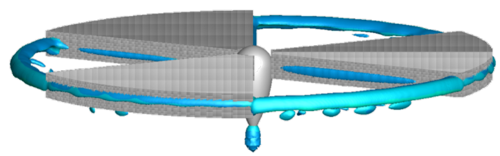
Timestep Convergence



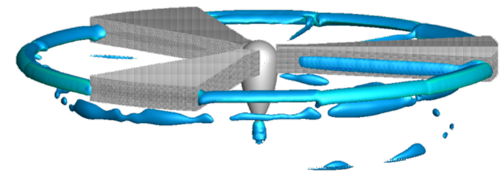
Timestep Convergence – Wake Structure



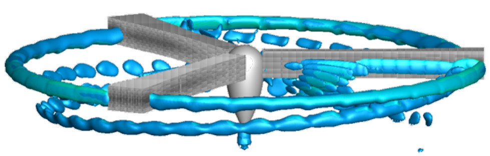
$\Omega\Delta t = 120$ deg



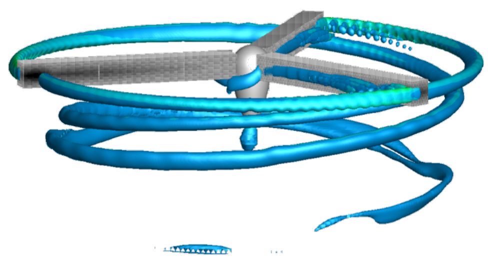
$\Omega\Delta t = 60$ deg



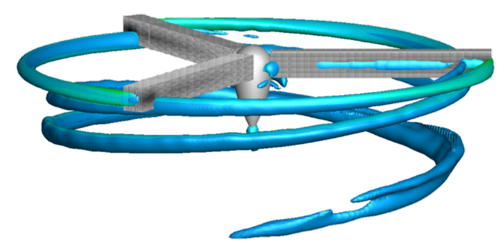
$\Omega\Delta t = 30$ deg



$\Omega\Delta t = 10$ deg



$\Omega\Delta t = 3$ deg



$\Omega\Delta t = 1$ deg

Robust & Performant Algorithms

Robust & Performant Algorithms

Important to have quick turnaround in preliminary design without significant user intervention each design cycle

- Robust algorithms avoid the need to continually monitor the solution process
- Parallel algorithms are required to leverage the continuing increase in modern processor core counts

In order to ensure robustness and performance throughout the simulation process, several algorithms were developed or analyzed in this work:

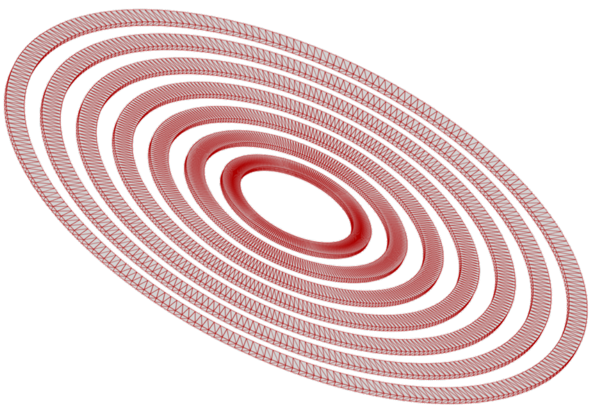
- Fast search for hexahedra
- Computation of source terms
- Time integration methods
- Airfoil table generation

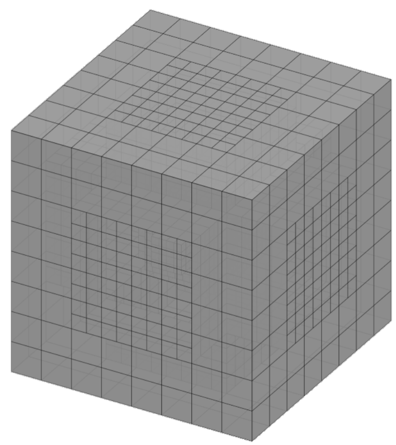
Fast Search for Rotor Hexahedra

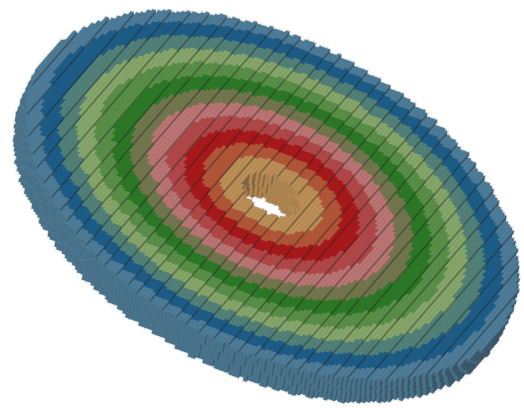
Rotating source terms through fixed mesh allows the search for rotor hexes to be done once as a preprocessing step

- Triangulate each blade element as a cylindrical shell
- Leverage rapid triangle-cube intersection algorithms in Cart3D framework

No neighbor information is required, process cells in parallel





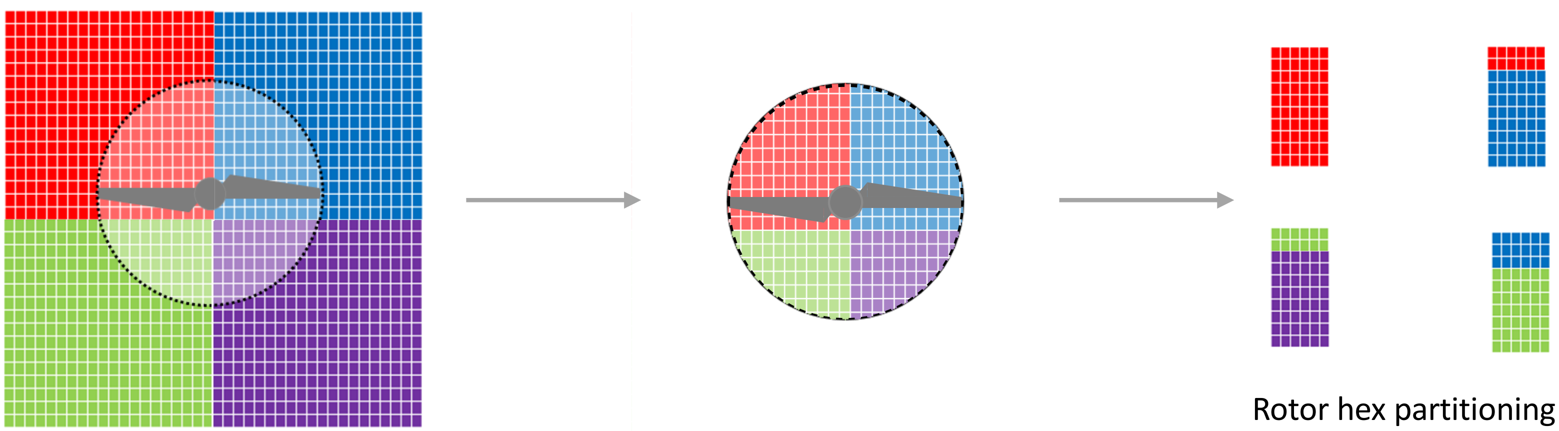


Rotor Hexahedra Partitioning

Only a fraction of the cells will receive rotor source terms

Existing domain decomposition in Cart3D assigns approximately equal work to each partition

Assign similar amounts of rotor modeling work to each partition and communicate the source terms back

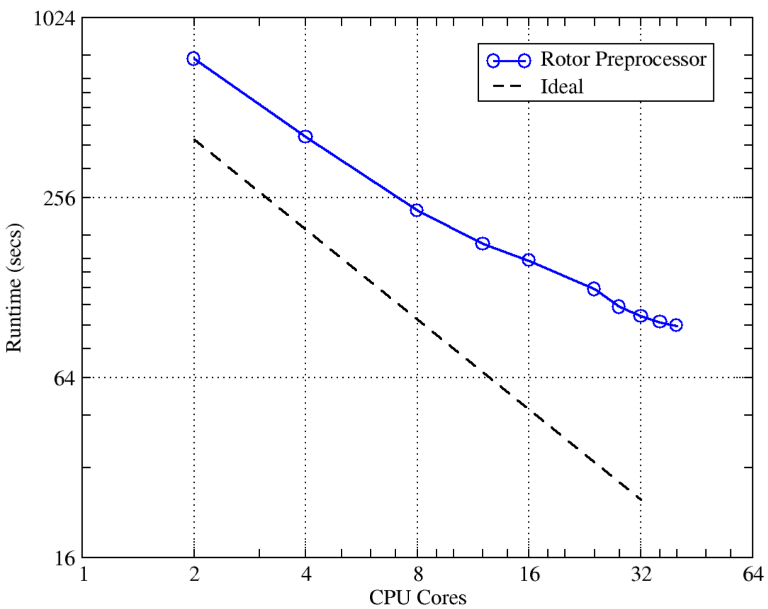


Rotor hex partitioning

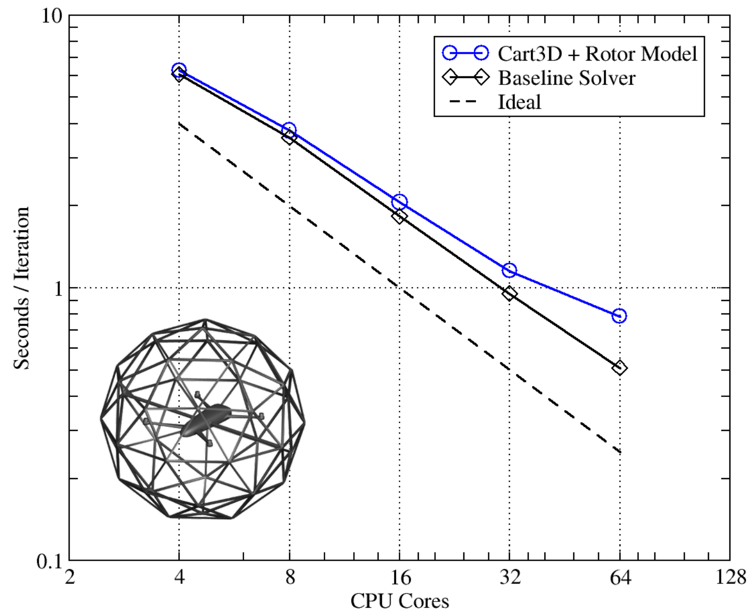
Domain partitioning

Strong Scaling Performance

Rotor Preprocessor



Rotor Model + Flow Solver



Time Integration Methods

General multi-rotor aircraft simulations are time-dependent

Implicit time integration methods alleviate the small timestep restriction imposed by explicit methods

Dual time stepping is commonly used to solve the implicit equations (Jameson, 1991)

$$\frac{d\mathbf{u}}{dt} - \lambda \mathbf{u} = 0$$

Model problem

$$\frac{\partial \mathbf{u}}{\partial \tau} + \mathcal{D}_t \mathbf{u} - \lambda \mathbf{u} = 0$$

Dual time stepping equations

Dual Time Stepping

The stability properties of the implicit temporal operator are only guaranteed if the unsteady residual is driven to zero

Unfortunately, this tends to be prohibitively expensive even on modern HPC platforms and common best practice is to reduce the residual by 2-3 orders of magnitude

Does partial convergence affect solution robustness?

Perform linear stability analysis to investigate this effect

Implicit Euler with Dual Time Stepping

1st order implicit Euler with explicit driving scheme

$$\frac{\partial \mathbf{u}}{\partial \tau} \quad + \quad \mathcal{D}_t \mathbf{u} \quad - \lambda \mathbf{u} = 0$$

$$\frac{\mathbf{u}^{n+1,k+1} - \mathbf{u}^{n+1,k}}{\Delta \tau} + \frac{\mathbf{u}^{n+1,k+1} - \mathbf{u}^{n}}{\Delta t} - \lambda \mathbf{u}^{n+1,k} = 0$$

Multiply by Δt and let γ be the ratio of timesteps $\Delta t/\Delta\tau$

$$\mathbf{u}^{n+1,k+1} = \frac{\gamma + \lambda \Delta t}{\gamma + 1} \mathbf{u}^{n+1,k} + \frac{1}{\gamma + 1} \mathbf{u}^{n}$$

Implicit Euler with Dual Time Stepping

Assume a natural initial condition

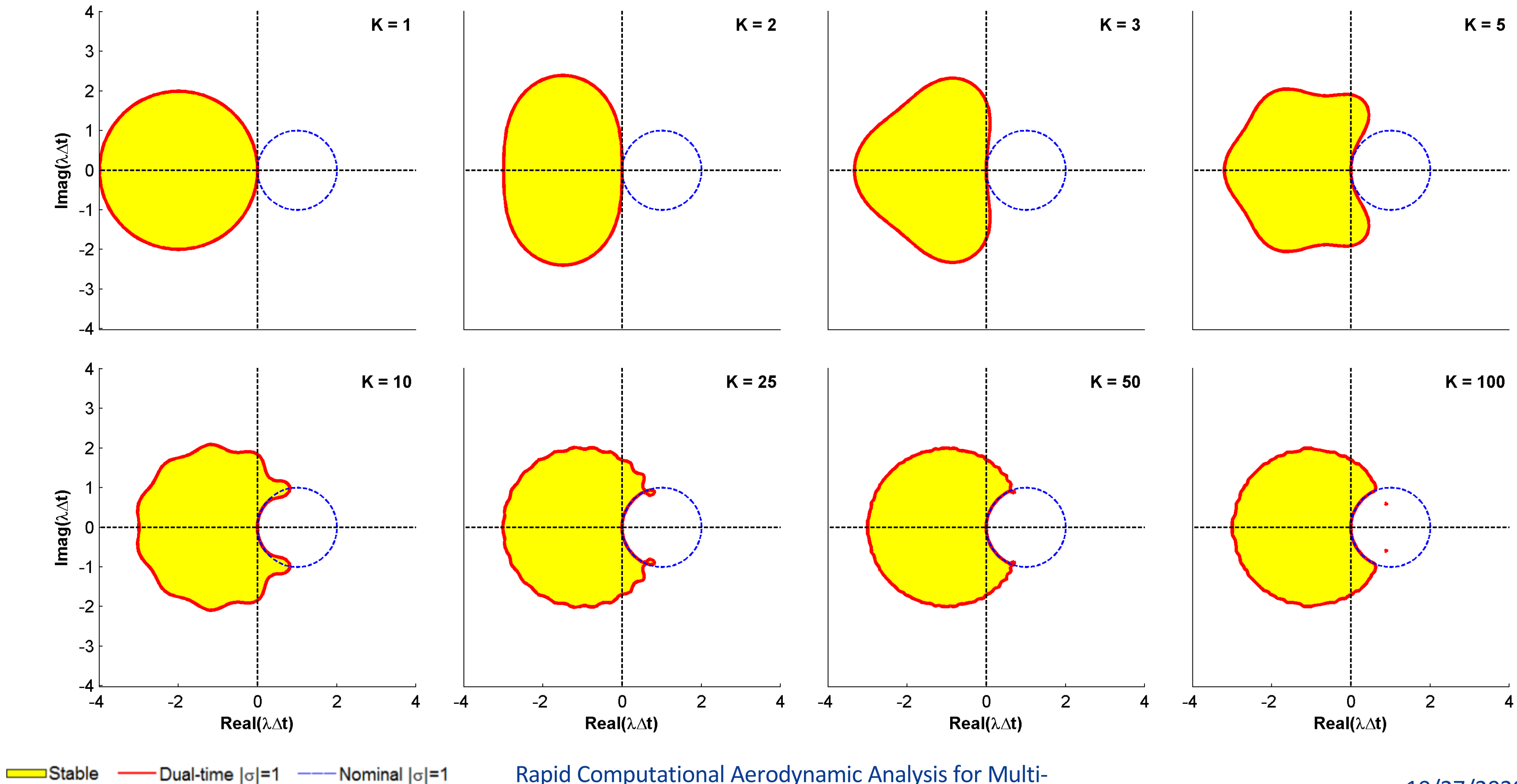
$$\mathbf{u}^{n+1,0} = \mathbf{u}^n$$

Divide through by $\mathbf{u}^n$

$$\frac{\mathbf{u}^{n+1,K}}{\mathbf{u}^n} = \left(\frac{\gamma + \lambda\Delta t}{\gamma + 1}\right)^K + \frac{1}{\gamma + 1}\sum_{j=0}^{K-1}\left(\frac{\gamma + \lambda\Delta t}{\gamma + 1}\right)^j$$

We can plot this in the complex plane and determine the stability limits of this method

Implicit Euler with Dual Time Stepping

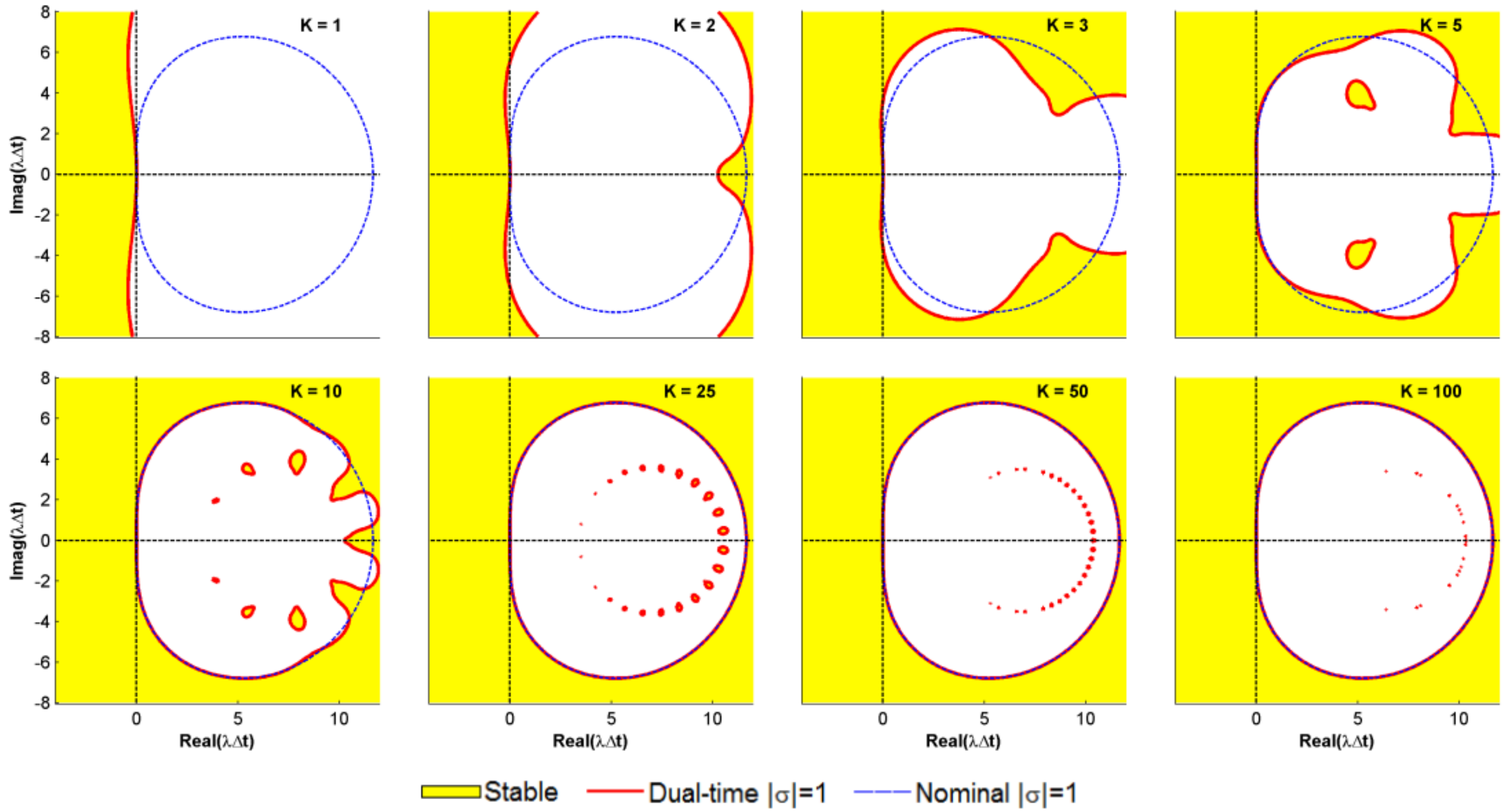


Summary of Linear Analysis

Performed linear stability analysis for two backwards difference schemes as well as general diagonally implicit Runge-Kutta methods

Even 2nd order methods can potentially lose A-stability with insufficient convergence, but the effect is more pronounced with high-order, multi-stage methods

SDIRK2 - Implicit residual evaluation



Summary of Linear Analysis

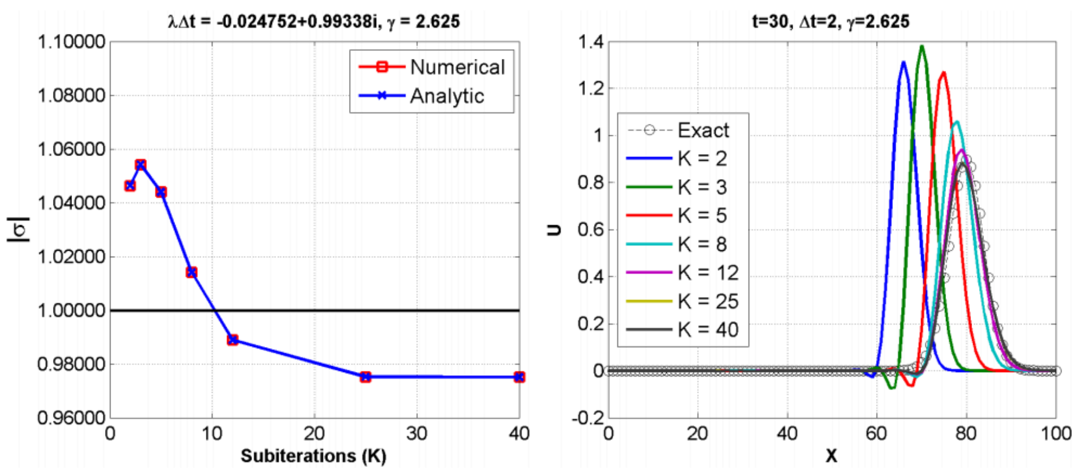
Performed linear stability analysis for two backwards difference schemes as well as general diagonally implicit Runge-Kutta methods

Even 2nd order methods can potentially lose A-stability with insufficient convergence, but the effect is more pronounced with high-order, multi-stage methods

How does this affect actual numerical simulations?

Linear Advection-Diffusion Example

Perform numerical simulations using linear advection and advection-diffusion to both confirm the theoretical results and demonstrate the effect of subiteration convergence



(a) Amplification factor vs K (b) Final solution vs. K

SDIRK2: Advection-Diffusion of approximate Gaussian for 15 steps, $b = \frac{1}{20}, \Delta t = 2$, 101 pts

Summary of Linear Analysis

Performed linear stability analysis for two backwards difference schemes as well as general diagonally implicit Runge-Kutta methods

Even 2nd order methods can potentially lose A-stability with insufficient convergence, but the effect is more pronounced with high-order, multi-stage methods

How does this affect actual numerical simulations?

In all cases studied, the stability criterion was satisfied faster than the accuracy criterion, so the impact is minimal

Robust Airfoil Table Generation

Blade Element Theory methods rely on airfoil tables to compute accurate propeller performance

Low Reynolds number aerodynamics for small propellers requires reasonable boundary-layer transition predictions

Numerically generated airfoil tables can fail to converge (Russell, 2017)

Developed an automated, robust airfoil table generator and implemented a novel smoother based on 1-D limiters

Robust Airfoil Table Generation

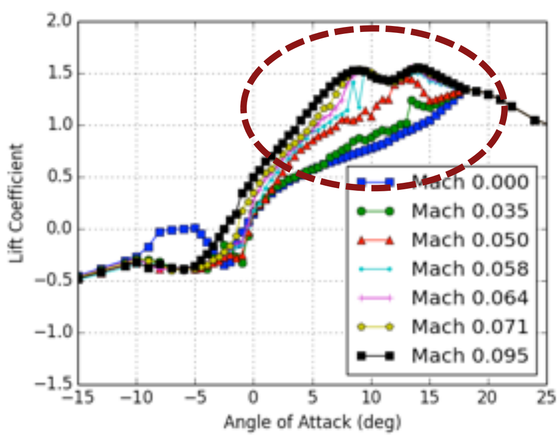
Use thin airfoil theory results to ensure full coverage

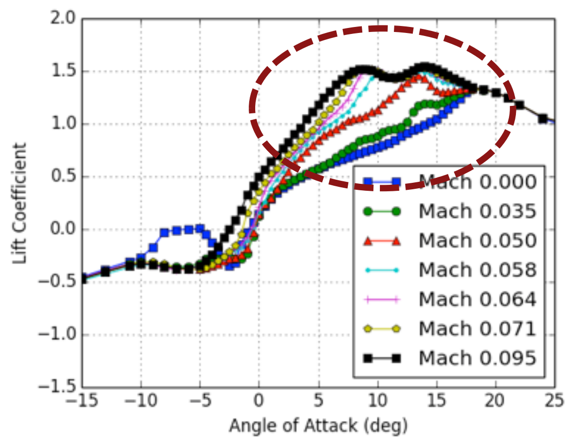
Apply any specified experimental data

Run XFOIL with variety of panelizations (60-100), which helps ensure convergence, and take best available solution

Smooth table with van Leer limiter as detector, take average

Raw





Smoothed

Model Validation

XV-15 Hover

3 bladed proprotor

NACA 64-XXX airfoil sections

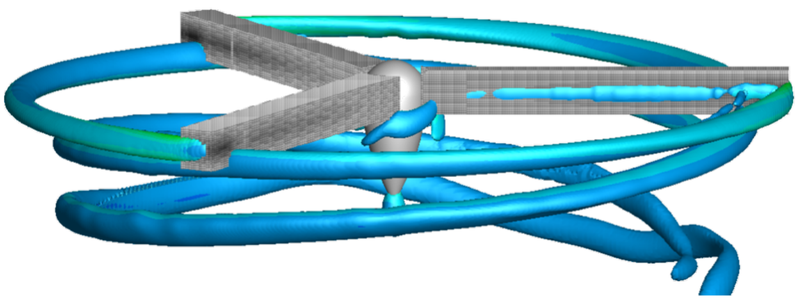
Tip Mach = 0.69

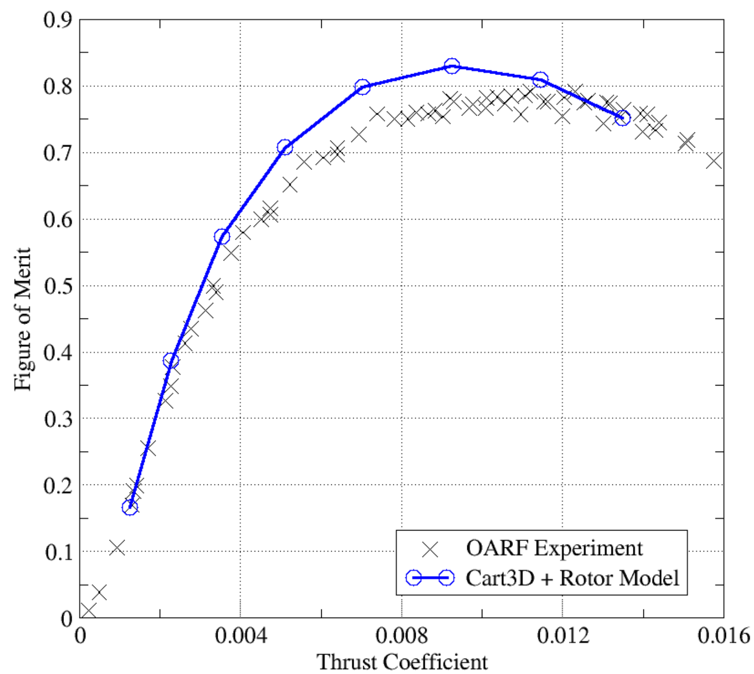
-41° twist

Compare to OARF hover data for isolated rotor

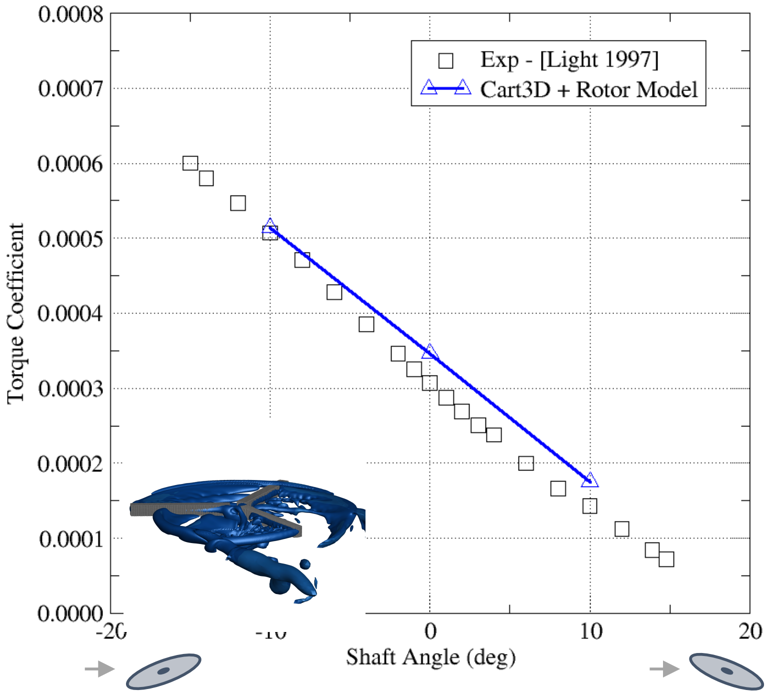
Use existing NASA airfoil tables

XV-15 Hover




0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
Figure of Merit
0
0.004
0.008
0.012
0.016
Thrust Coefficient
OARF Experiment
Cart3D + Rotor Model

XV-15 Edgewise Flight



Assess model in edgewise forward flight with more complicated rotor inflow

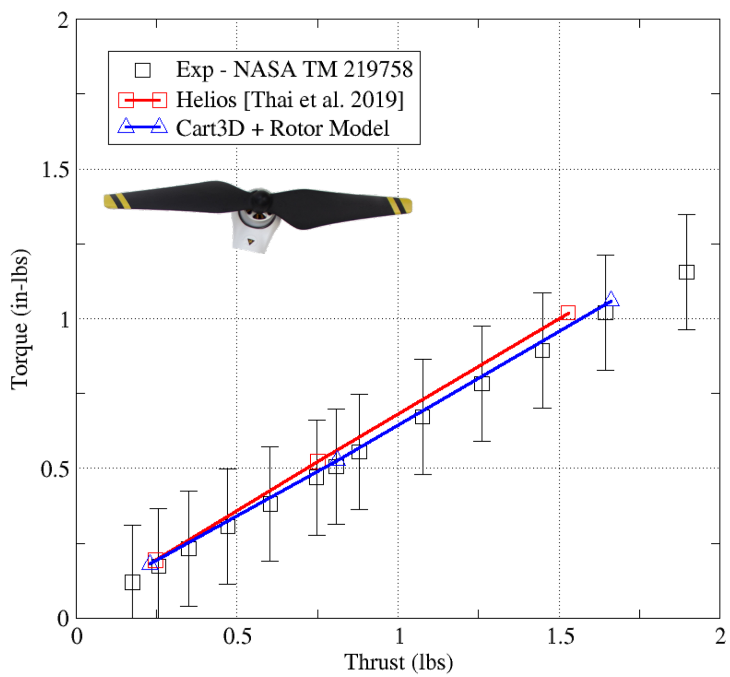
Compare to NASA wind tunnel test data (Light, 1997)

Rotor is trimmed to constant thrust ($C_T/\sigma = 0.075$)

Mach number = 0.11

Free-air simulations without Rotor Test Apparatus (RTA)

DJI Phantom 3 Hover



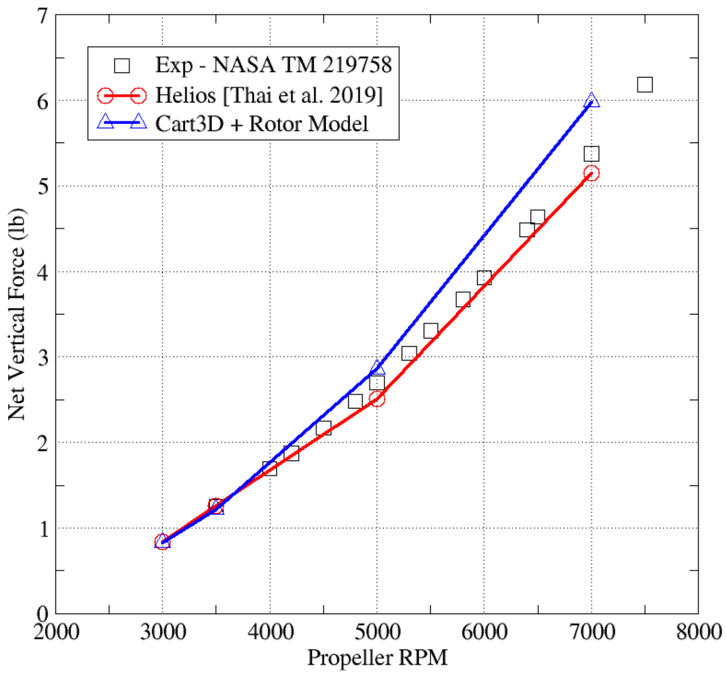
Isolated fixed pitch propeller from popular quadcopter

Tested in US Army 7x10 wind tunnel at NASA Ames in 2016

Navier-Stokes (SA-DES) simulations by Thai et al. (2019)

Compare hover performance data to validate rotor model with newly generated airfoil tables for low Reynolds number propeller

DJI Phantom 3 – Full Vehicle - Hover



Summary of Main Contributions

Developed scalable rotor model that accurately predicts rotor performance with rapid turnaround (~25x Helios)

- Verified and validated IVS for estimating AoA and Mach number
- Demonstrated need for interpolation to accurately capture spanwise loading
- Detailed space-time conservation approach to enable large timesteps
- Created a robust airfoil table generator suitable for low Reynolds number propellers

Derived linear dual time stepping amplification factor for several time integration methods and showed that the loss of stability, while possible, is not generally worrisome

Acknowledgments

National Innovation Center
National Land Imaging Program

NASA Aeronautics Research Mission Directorate
Transformational Tools & Technologies Project

Computer time was provided by the
NASA Advanced Supercomputing Division

Thoughtful discussions and insightful comments:

- Cart3D development team
- Philippe Spalart
- Austin Thai
- Chris Silva & Carl Russell

References

Rajagopalan, R. G., & Lim, C. K. (1991). Laminar flow analysis of a rotor in hover. *Journal of the American Helicopter Society*, *36*(1), 12-23.

Shen, W. Z., Hansen, M. O., & Sørensen, J. N. (2009). Determination of the angle of attack on rotor blades. *Wind Energy: An International Journal for Progress and Applications in Wind Power Conversion Technology*, *12*(1), 91-98.

Forsythe, J. R., Lynch, E., Polsky, S., & Spalart, P. (2015). Coupled flight simulator and CFD calculations of ship airwake using Kestrel. In *53rd AIAA Aerospace Sciences Meeting.*

Merabet, R., & Laurendeau, E. (2019). Parametric study on the velocity sampling techniques for the actuator line method in 2D. In *AIAA Scitech 2019 Forum.*

Thai, A., Jain, R., & Grace, S. (2019, January). CFD Validation of Small Quadrotor Performance using CREATE-AV Helios. In *VFS 75th Annual Forum & Technology Display*.

Russell, C. R., & Sekula, M. K. (2017). Comprehensive analysis modeling of small-scale UAS rotors. In *AHS 73rd Annual Forum.*

References

Koning, W. J. (2016). Wind Tunnel Interference Effects on Tilt Rotor Testing Using Computational Fluid Dynamics. NASA/CR—2016–219086.

Wissink, A., Lakshminarayan, V., Jude, D., Jayaraman, B., & Sitaraman, J. (2020). Assessment of Different CFD Download Predictions with Helios. In *VFS Transformative Vertical Flight 2020.*

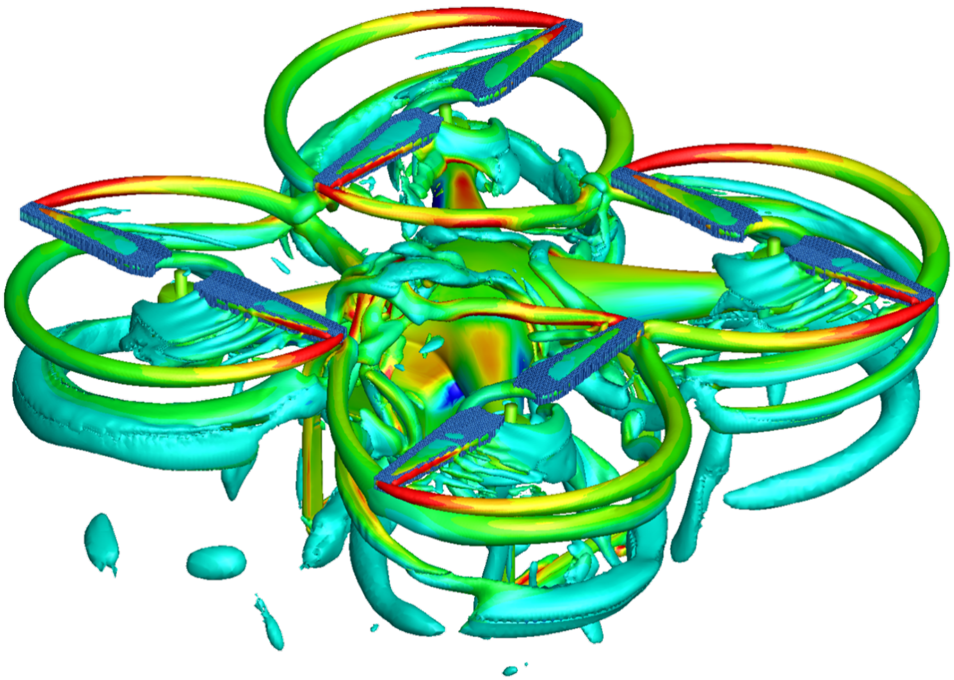
Light, J. S. (1997). Results from an XV-15 rotor test in the National Full-scale Aerodynamics Complex. In *AHS 53rd Annual Forum* (pp. 231-239).

Felker, F. F., Betzina, M. D., & Signor, D. B. (1985). Performance and loads data from a hover test of a full-scale XV-15 rotor. NASA/TM 86833.

Russell, C. R., Jung, J., Willink, G., & Glasner, B. (2016). Wind tunnel and hover performance test results for multicopter UAS vehicles. NASA/TM-2018–219758.

Jameson, A. (1991, June). Time dependent calculations using multigrid, with applications to unsteady flows past airfoils and wings. In *10th Computational Fluid Dynamics Conference*.

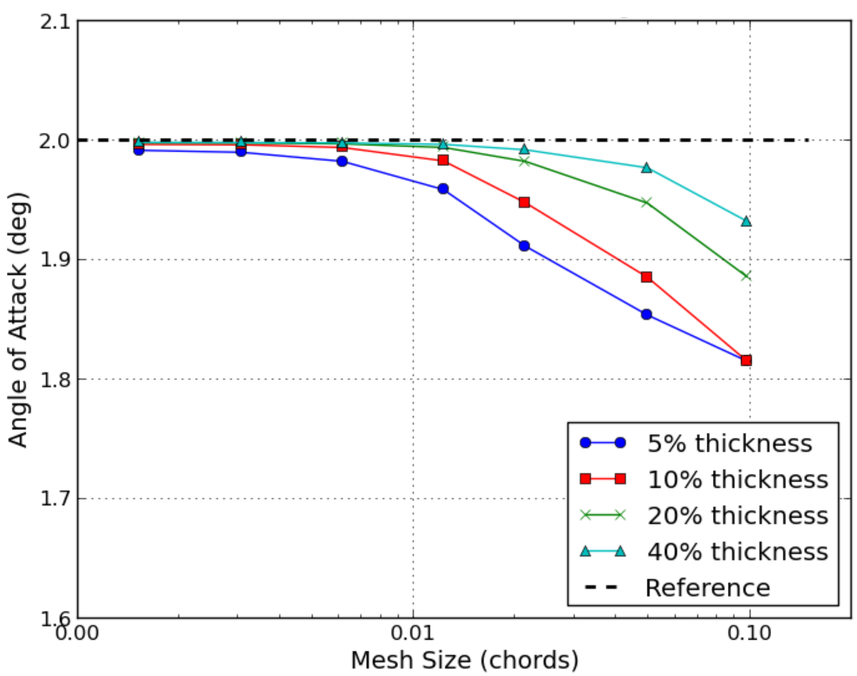
Thank You!



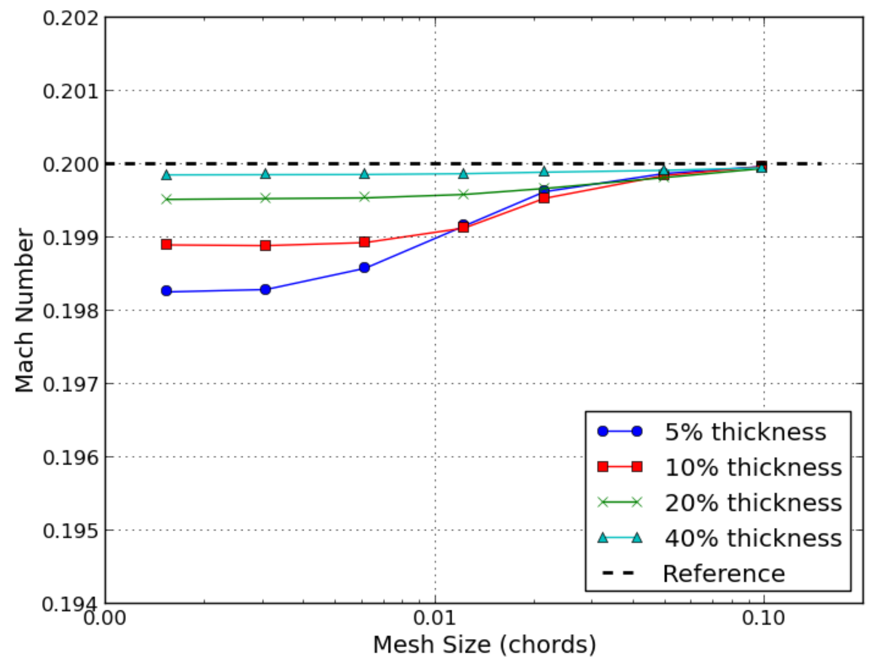
Questions?

IVS 2–D Accuracy: Zero Drag

Angle of Attack

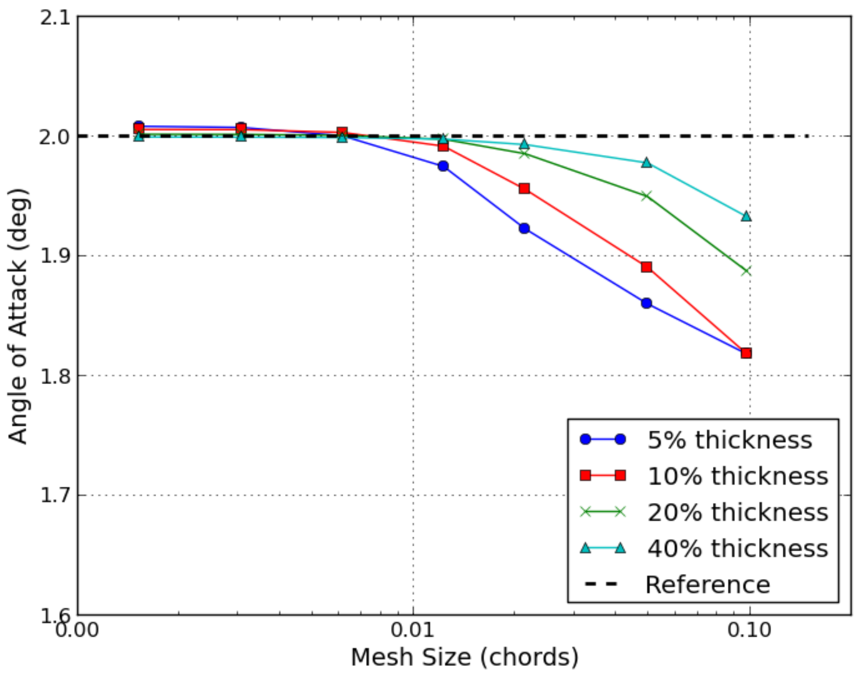


Mach Number

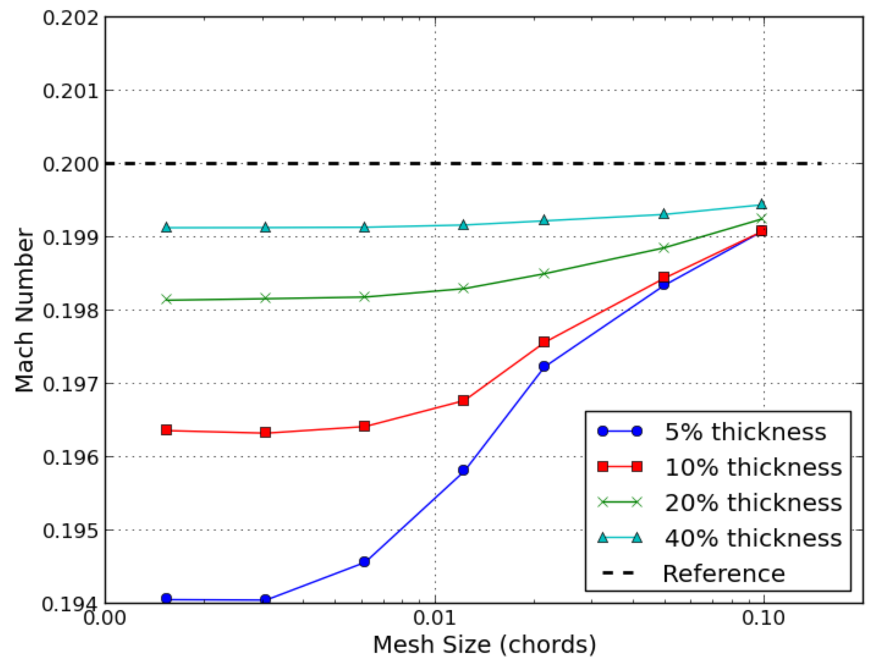


IVS 2–D Accuracy: With Drag

Angle of Attack



Mach Number

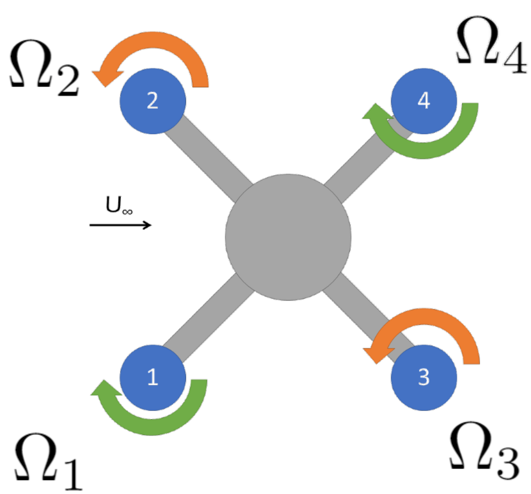


Multi-Rotor Trim

Trim is required for accurate forward flight predictions

Some examples in literature for both helicopters (Yang 2002) and multirotors (Thai 2020)

Use Newton's Method with pivoting LU solver and frozen flowfield for quadcopters



$$\begin{bmatrix} C_W \\ C_{RM} \\ C_{PM} \\ C_Q \end{bmatrix}^i + \begin{bmatrix} \frac{\partial C_W}{\partial \Omega_1} & \frac{\partial C_W}{\partial \Omega_2} & \frac{\partial C_W}{\partial \Omega_3} & \frac{\partial C_W}{\partial \Omega_4} \\ \frac{\partial C_{RM}}{\partial \Omega_1} & \frac{\partial C_{RM}}{\partial \Omega_2} & \frac{\partial C_{RM}}{\partial \Omega_3} & \frac{\partial C_{RM}}{\partial \Omega_4} \\ \frac{\partial C_{PM}}{\partial \Omega_1} & \frac{\partial C_{PM}}{\partial \Omega_2} & \frac{\partial C_{PM}}{\partial \Omega_3} & \frac{\partial C_{PM}}{\partial \Omega_4} \\ \frac{\partial C_Q}{\partial \Omega_1} & \frac{\partial C_Q}{\partial \Omega_2} & \frac{\partial C_Q}{\partial \Omega_3} & \frac{\partial C_Q}{\partial \Omega_4} \end{bmatrix} \begin{bmatrix} \Delta\Omega_1 \\ \Delta\Omega_2 \\ \Delta\Omega_3 \\ \Delta\Omega_4 \end{bmatrix} = \begin{bmatrix} C_{W,tgt} \\ 0 \\ 0 \\ 0 \end{bmatrix}$$